Data Compression for NGST

María A. Nieto-Santisteban, Dale J. Fixsen[1], Joel D. Offenberg[1], Robert J. Hanisch, H. S. (Peter) Stockman.

Space Telescope Science Institute, 3700 San Martin Drive, MD 21218
Email: nieto, fixsen, offenbrg, hanisch, stockman@stsci.edu
URL: http://ngst.gsfc.nasa.gov/cgi-bin/iptsprodpage?Id=14

Abstract. The immense amount of data that the Next Generation Space Telescope (NGST) will produce and its distant orbit from Earth make it mandatory to do some amount of on-board image processing and data compression.

This paper gives a summary of the performance of several lossless compression methods. We also show results of prescaling the image prior to compression using a square-root function. This imposes a slightly lossy compression, but the scaling can be adjusted so as to retain the desired number of noise bits.

1. Introduction

The Next Generation Space Telescope (NGST) will produce about 600 GB/day, assuming we use the NASA Yardstick 8k × 8k NIR camera (16 bits/pixel), save and transmit 64 non-destructive read-outs per image, and the camera is in continuous use (about 80 observations/day, 10^3 s each). However, with an L2 halo orbit, the NASA NGST study estimates a downlink rate of 5.35 GB/day using X-band. Clearly the volume of data to downlink must be reduced by at least a factor of 100.

Astronomical images are noisy. This fact makes them difficult to compress by lossless compression algorithms such as Huffman, Lempel-Ziv, run-length, or arithmetic code. However, they also have the virtue of showing similar values among adjacent pixels. Techniques such as Rice's algorithm (Rice, Yeh & Miller 1993) and derivatives (White & Becker 1998; Stiavelli & White 1997) can take advantage of this. In this paper, we present how some of these compression techniques would work with NGST images. Unfortunately, these lossless algorithms give us compression ratios that still exceed the telemetry guidelines. We have also looked into the feasibility of doing lossy compression by scaling the original image prior to the lossless compression. Under this scheme, we find substantial data reduction with a negligible effect on the data quality.

---

[1]Raytheon STX Corporation, 4400 Forbes Blvd, Lanham, MD 20706

2. The Data Compression Process: Ratios

The first and more important compression ratio, **64:1**, is obtained after applying a cosmic ray rejection process and fitting 64 readouts into one single image (Offenberg et al. 1999). (In fact, 65 readouts involve the cosmic ray rejection and fitting process. However, the first readout (dark frame) should be rarely downlinked.)

An additional compression factor close to **3:1** is achieved by using predictive compression techniques such as the Lossless JPEG and Rice's algorithm. Dictionary-based lossless compression programs such as "gzip" and "compress" present lower compression ratios (see Table 1).

We can further reduce the data volume by using a prescaled image as input to the lossless encoder. The prescaling process, based on the square-root function, can be adjusted so as to retain as many bits of noise as desired. Similar results are obtained independently of the lossless compression technique in use, with overall compression ratios of **4:1** (keeping 4 noise bits), up to as much as **8:1** (keeping 1 noise bit).

Table 1. Performance Summary

Compressor	Time (s)	Memory (KB)	Lossless (bpp)	Lossy (bpp) $N_b = 1$	$N_b = 2$	$N_b = 3$	$N_b = 4$
rcomp	0.253	2904	5.473	2.104	3.087	3.612	3.997
uses	0.737	4040	5.477	2.107	2.988	3.529	3.952
LLjpeg	2.539	4152	5.341	2.097	2.933	3.431	3.877
gzip	4.871	1112	6.369	2.125	3.233	3.886	4.425
compress	0.747	1600	6.036	1.691	2.762	3.474	4.015
RMS Error				0.510	0.252	0.186	0.123
Mean Diff.				-0.079	-0.024	-0.078	0.007

This table give the summary in time, memory and compression rates (lossless and lossy) achieved for five **lossless compressors**. As test input data, a simulated NGST deep image was used (1024×1024, 2 bytes per pixel, DN units) obtained in a 10^3s exposure after the cosmic ray removal and slope fitting algorithm. A readout noise of 15 electrons and a gain of 4 are assumed. Scaling function applied for lossy compression, $N_B \sqrt{G} \sqrt{D + Y'}$. Values of the Normalized Root-Mean-Square Error and Mean Difference are also shown. *Rcomp* and *uses* are implementations of the Rice algorithm. *Rcomp* was developed at STScI by Rick White. *Uses* was developed at University of New Mexico Microelectronics Research Center. *LLjpeg* is a lossless JPEG developed by the PVRG-JPEG. *gzip* and *compress* are the well known general purpose compression programs based on dictionary techniques. The tests were run on a Sun Ultra 10.

3. Lossy Compression: Square-Root Prescaling

Pure noise by its very nature is impossible to compress. In order to compress the data and retain as much information as possible, it is useful to eliminate the low order data bits (i.e., the noise). Truncating the bits at some level is one possibility, but the noise level differs across the picture. We describe a simple approximation to the noise which is carried in the data itself.
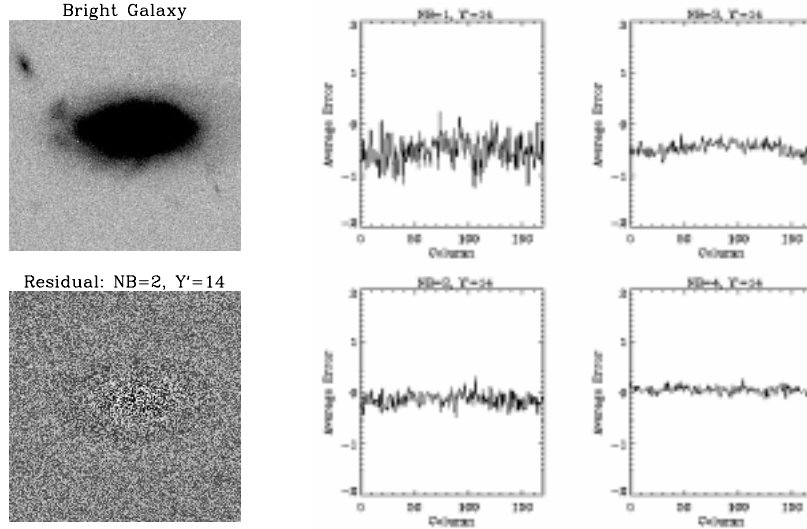
Figure 1.   On the left, a bright galaxy from a simulated NGST long exposure. On the right, residual plots varying the number of bits into the noise.

There are two important sources of non-systematic noise: the Poisson distribution of the photons themselves and the readout noise in the readout electronics. From these two sources, a reasonable approximation to the standard deviation is $\sigma = \sqrt{S + R}$ where S is the signal in units of photons (or equivalently electrons) and R is the readout variance in the same units.

In order to simplify both the encoding and the decoding we make an approximation to $\sigma \simeq \sqrt{S + Y}$ where S is the readout (in electrons) and Y is a number which we will derive from the readout variance R. First note an absolute offset does not incur a significant penalty in data transmission since virtually any lossless data compression scheme will use very little bandwidth to transmit the offset in a large block of data. The key then is to get a good approximation to the derivative. Note if we use $\sqrt{S + Y}$ for $S/\sqrt{S + R}$ for large S the derivatives already match. By taking the derivatives with respect to S and setting them equal to each other at $S = 0$ we have $Y = R/4$. Thus we can use $\sqrt{S + R/4}$ and truncation as a lossy compression to reduce the data dynamic range in an approximately uniform way. Since our NGST simulation is in data numbers (DN), we must rescale this formula to $\sqrt{G}\sqrt{D + Y'}$, where G is the gain, D is the signal in units of DN and $Y' = R/4G$. Therefore, assuming a 15 electrons readout noise and a gain of 4, $Y' = 14$. (Note $R = 15^2$).

The remaining issue is where to do the truncation. We multiply by a value N_B and round to the nearest integer. The value N_B is then the number of bits into the noise that we save. Setting N_B to 1 has the effect of severely restricting the noise at each pixel but it also may affect averages and fits that attempt to pull a signal out of the noise. Larger values of N_B allow the compressed data to more closely match the original values, but at less effective compression ratios (see RMS Error in Table 1). We suggest using $N_B \sim 2$, a value that gets us close to but under the $1/\sqrt{12}$ that we could expect from the digitization noise. Figure 1 shows how this lossy scheme does not introduce any pattern or systematic error into the residual images. The small bias shown is due to the digitization process. If this were a problem in practise, we could add a random bias which will (on average) remove it.

3

4.   Compression Benefits from Cosmic Ray Removal

For the the maximum exposure time we have adopted, 10^3 s, it is expected that 10% of the image is affected by cosmic rays hits (Stockman et al. 1998). Most of the cosmic rays produce high values in one single pixel unrelated to other values in the vicinity. All tested compression programs benefit from prior cosmic ray removal. The Rice implementation used in this paper utilizes a linear first-order unit-delay predictor whose output is equal to the difference between the input data value and the preceding data value (CCSDS 1997). This algorithm particularly benefits from CR removal.

5.   Conclusions

- It is possible to reduce by more than two orders of magnitude the anticipated volume of data.
- Our lossy method does not introduce any important distortion into the data. The prescaling may be very useful for scenes with high dynamic range, such as 2-D spectra or images of dense star clusters.
- Compression benefits from prior cosmic ray rejection.
- According to the Performance Summary (Table 1) Rice would be our choice of compression. It is the fastest algorithm. The bit rates given by Rice are among the best. Current implementations require the most memory, but this can be improved. The Consultative Committee for Space Data Systems (CCSDS) recommends Rice for Lossless Data Compression.

References

Consultative Comittee for Space Data System 1997, CCSDS120.0-G-1 Green Book.

Offenberg, J. D., Sengupta, R., Fixsen, D. J., Stockman, H. S. (Peter), Nieto-Santisteban, M. A., Stallcup, S., Hanisch, R. J. & Mather, J. C. 1999, this volume, [P4.8].

Rice, R. F., Yeh, P.S. & Miller, W. H. 1993, in Proceedings of the AIAA Computing in Aerospace 9 Conference (AIAA-93-4541-CP).

Stiavelli, M. & White, R. L. 1997, STScI Instrument Science Report ACS-97-02.

Stockman, H. S. (Peter), Fixsen, D. J., Hanisch, R. J., Mather, J. C., Nieto-Santisteban, M. A., Offenberg, J. D., Sengupta, R. & Stallcup, S. 1998, astro-ph/9808051.

White, R. L. & Becker, I. 1998, in Proc. SPIE, eds P. Y. Bely & J.B. Breckinridge, v.3356, p 823-831.